Mapping Repair in Ontology-based Data Access Evolving Systems

Domenico Lembo[1], **Riccardo Rosati**[1], **Valerio Santarelli**[1],
Domenico Fabio Savo[1], **Evgenij Thorstensen**[2]

[1] Sapienza Università di Roma
[2] University of Oslo
[1] ⟨lastname⟩@dis.uniroma1.it, [2] evgenit@ifi.uio.no

Abstract

In this paper we study the evolution of ontology-based data access (OBDA) specifications, and focus on the case in which the ontology and/or the data source schema change, which may require a modification to the mapping between them to preserve both consistency and knowledge. Our approach is based on the idea of repairing the mapping according to the usual principle of minimal change and on a recent, mapping-based notion of consistency of the specification. We define and analyze two notions of mapping repair under ontology and source schema update. We then present a set of results on the complexity of query answering in the above framework, when the ontology is expressed in *DL-Lite$_R$*.

1 Introduction

Ontology-based data access (OBDA) is the problem of accessing source databases through the mediation of a conceptual domain view, given in terms of an ontology [Poggi *et al.*, 2008]. A major issue in OBDA concerns the design of a specification and the management of its evolution. An *OBDA specification* is constituted by an ontology, usually a Description Logic (DL) TBox [Baader *et al.*, 2007], a schema of the source databases, and a mapping specifying the relationship between the data at the sources and the elements of the ontology, which is commonly given as a set of assertions, each one associating a query over the source schema with a query over the ontology. In the following we denote an OBDA specification as $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$, where $\mathcal{T}$ is the TBox, $\mathcal{S}$ is the source schema, and $\mathcal{M}$ is the mapping.

The mapping is certainly the component of the specification that has received so far less attention, and thus consolidated tools for its design and maintenance are currently not available. Mapping design is a time-consuming and complex operation, which in general cannot be totally automatized. Of course, modifying the mapping due to changes in the other components of the specification may result in a time-consuming process as well. We experienced this in various industrial and academic projects. Among them, we mention a collaboration with the Italian Ministry of Economy and Finances, where we had to map a domain ontology with a source database completely independent from it, which caused mapping definition

to be particularly complex [Antonioli *et al.*, 2014], and the use cases of the EU project Optique, focused on OBDA for Big Data [Giese *et al.*, 2015], which were particularly challenging with respect to mapping design and analysis.

In this paper we study the evolution of OBDA specifications. We first observe that many approaches exist for both *ontology evolution* [Zablith *et al.*, 2015] and *database schema evolution* [Rahm and Bernstein, 2006]. However, to the best of our knowledge, no previous study has analyzed evolution in the presence of a mapping connecting an ontology to a database schema. In this sense, a problem that is close to OBDA is *ontology matching and alignment*, which is based on the use of a notion of mapping to integrate different ontologies. Several works have studied the problem of repairing inconsistent mappings in this context (e.g., [Jiménez-Ruiz *et al.*, 2013; Meilicke *et al.*, 2009; Qi *et al.*, 2009]). However, the framework of ontology matching, and in particular the notion of mapping, is very different from OBDA.

We adopt a *mapping-centered* notion of OBDA evolution: given an OBDA specification $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$, we want to repair the mapping $\mathcal{M}$ given a modification of the TBox $\mathcal{T}$ and/or of the source schema $\mathcal{S}$. This is a natural assumption: indeed, the mapping is an information that depends on both the TBox and the source schema, while the TBox and the schema are (at least in principle) semantically independent entities (since the data sources are autonomous systems).

Following the classical approaches to belief revision, we want to find a notion of repair of a mapping that is based on two general principles: (i) it should preserve *consistency* of the OBDA specification; (ii) it should express *minimal change* with respect to the initial OBDA specification. With respect to consistency preservation, we adopt a non-classical notion of inconsistency for an OBDA specification, called *global mapping inconsistency*, recently introduced in [Lembo *et al.*, 2014; 2015]. According to this notion, a mapping $\mathcal{M}$ is inconsistent with respect to a TBox $\mathcal{T}$ and a source schema $\mathcal{S}$ if there exists no instance D for $\mathcal{S}$ that *activates* all the mapping assertions in $\mathcal{M}$, i.e., every query over $\mathcal{S}$ appearing in $\mathcal{M}$ has a non-empty answer on D. In the context of OBDA, global mapping inconsistency provides a more meaningful notion of inconsistency than the classical one (which indeed considers all possible source instances): for example, in all the cases when $\mathcal{S}$ is a relational schema with standard integrity constraints, the OBDA specification is inconsistent according

to the classical semantics iff its TBox is inconsistent (which in turn implies that this notion is trivial for many DLs).

With respect to minimal change, we propose two different notions of repair. The first one, called *deletion-based mapping repair (DMR)*, reflects the simple idea of repairing a mapping through a (subset-)minimal deletion of assertions from the initial mapping. The second notion, called *entailment-based mapping repair (EMR)*, relies on the *mapping entailment set (MES)*. The MES of an OBDA specification $\mathcal{J}$ for a mapping language $\mathcal{L}$ is a set of mapping assertions in $\mathcal{L}$ that are logical consequences of $\mathcal{J}$. Then, the repairs are globally consistent mappings that allow to preserve as much as possible of the initial MES, according to a minimality criterion that formalizes the intuitive principle of preferring insertions over deletions.

Finally, we define and study query entailment under mapping repairs. In particular, we focus on TBoxes specified in *DL-Lite$_R$* [Calvanese *et al.*, 2007], the logical counterpart of the W3C standard OWL 2 QL [Motik *et al.*, 2009], which is a prominent DL in OBDA, and consider both GAV and GLAV mappings, two classical mapping languages used in data integration and OBDA [Doan *et al.*, 2012; Poggi *et al.*, 2008]. In these settings we provide a number of complexity results on the entailment of conjunctive queries.

2 Preliminaries

Source schemas. A source schema $\mathcal{S}$ is a relational schema, possibly equipped with integrity constraints (ICs) [Abiteboul *et al.*, 1995]. A *legal instance D for $\mathcal{S}$* is a database for $\mathcal{S}$ (i.e., a possibly infinite set of ground atoms over $\mathcal{S}$) that satisfies the ICs of $\mathcal{S}$. A source schema $\mathcal{S}$ is consistent if at least one legal instance for $\mathcal{S}$ exists. If $\mathcal{S}$ has no ICs, it is called *simple*. We adopt standard notions for conjunctive queries (CQs) over relational schemas and with $\phi(\vec{x})$ we denote a CQ with free variables $\vec{x}$. A Boolean CQ (BCQ) is a CQ without free variables. Given a CQ q over $\mathcal{S}$ and a legal instance D for $\mathcal{S}$, $eval(q, D)$ denotes the evaluation of q over D. With a little abuse of notation, we will sometimes treat a CQ q as the set of the atoms in q.

Ontologies. A DL ontology represents knowledge through concepts, denoting sets of objetcs, and roles, denoting binary relationships between concepts, and is constituted by a TBox, i.e., a finite set of intensional assertions, and an ABox, i.e., a finite set of extensional assertions [Baader *et al.*, 2007]. In this paper we focus on *DL-Lite$_R$* [Calvanese *et al.*, 2007] TBoxes, whose assertions have the following forms

$$B_1 \sqsubseteq B_2 \qquad B_1 \sqsubseteq \neg B_2 \qquad \text{(concept inclusions)}$$
$$R_1 \sqsubseteq R_2 \qquad R_1 \sqsubseteq \neg R_2 \qquad \text{(role inclusions)}$$

where B_1 and B_2 are basic concepts, i.e., expressions of the form A, $\exists P$, or $\exists P^-$, and R_1 and R_2 are basic roles, i.e., expressions of the form P, or P^-, where A and P denote an atomic concept and an atomic role, respectively. Assertions in the left-hand side (resp. right-hand side) are called positive (resp. negative) inclusions. The semantics of DL ontologies is given in the standard way [Baader *et al.*, 2007], and as in the W3C standard OWL 2 [Hitzler *et al.*, 2012], we do not interpret ontologies under the Unique Name Assumption. Given a TBox $\mathcal{T}$, we denote with $Mod(\mathcal{T})$ the set of models of $\mathcal{T}$, and with $\mathcal{T} \models \alpha$ the fact that $\mathcal{T}$ entails a sentence or TBox

assertion α. A TBox is said *inconsistent* if $Mod(\mathcal{T}) = \emptyset$. A *CQ over $\mathcal{T}$* is a CQ over the signature of $\mathcal{T}$, i.e., the set of its atomic concepts and roles.

Mappings. A *mapping assertion m* from a source schema $\mathcal{S}$ to a TBox $\mathcal{T}$ has the form $\phi(\vec{x}) \rightsquigarrow \psi(\vec{x})$, where $\phi(\vec{x})$, called the *body of m* and denoted $body(m)$, and $\psi(\vec{x})$, called the *head of m* and denoted $head(m)$, are queries over $\mathcal{S}$ and $\mathcal{T}$, respectively, both with free variables $\vec{x}$, called *frontier variables*. A mapping $\mathcal{M}$ from $\mathcal{S}$ to $\mathcal{T}$ is a *possibly infinite* set of mapping assertions from $\mathcal{S}$ to $\mathcal{T}$. In this paper, we consider the notable cases in which $\phi(\vec{x})$ is a CQ, and $\psi(\vec{x})$ is either a CQ without constants (GLAV mapping language), or a single-atom query without constants and existential variables (GAV mapping language) [Doan *et al.*, 2012]. Both mappings are special cases of the W3C standard R2RML [Das *et al.*, 2012]. In the following, given a mapping language $\mathcal{L}$, with an $\mathcal{L}$-mapping we mean a possibly infinite subset of $\mathcal{L}$.

We say that a mapping assertion $\phi(\vec{x}) \rightsquigarrow \psi(\vec{x})$ *is active on a source instance D* if $eval(\phi(\vec{x}), D) \neq \emptyset$. A mapping $\mathcal{M}$ is active on D if each $m \in \mathcal{M}$ is active on D.

OBDA specifications. An OBDA specification is a triple $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$, where $\mathcal{T}$ is a TBox, $\mathcal{S}$ is a source schema, and $\mathcal{M}$ is a mapping between the two, all as aforementioned. The semantics of $\mathcal{J}$ is given with respect to a legal instance D for $\mathcal{S}$: a model for $\mathcal{J}$ w.r.t. D is a first-order interpretation $\mathcal{I}$ that satisfies $\mathcal{T}$ and *satisfies $\mathcal{M}$ w.r.t. D*, i.e., if for each assertion $\phi(\vec{x}) \rightsquigarrow \psi(\vec{x})$ in $\mathcal{M}$ and each tuple of constants $\vec{t}$ such that $\vec{t} \in eval(\phi(\vec{x}), D)$, we have that $\mathcal{I} \models \psi(\vec{t})$. The set of models of $\mathcal{J}$ w.r.t. D is denoted with $Mod(\mathcal{J}, D)$. Also, we use $(\mathcal{J}, D)$ to denote $\mathcal{J}$ with source instance D, say that $(\mathcal{J}, D)$ *is inconsistent* if $Mod(\mathcal{J}, D) = \emptyset$, and denote with $(\mathcal{J}, D) \models \alpha$ the entailment of a sentence or TBox assertion α by $(\mathcal{J}, D)$. Given a mapping assertion m, we say that $\mathcal{J} \models m$ if for every legal instance D for $\mathcal{S}$, and every $\mathcal{I} \in Mod(\mathcal{J}, D)$, $\mathcal{I}$ satisfies the mapping $\{m\}$ w.r.t. D. We also say that two mappings $\mathcal{M}_1$ and $\mathcal{M}_2$ from $\mathcal{S}$ are equivalent with respect to $\mathcal{T}$ if the specifications $\langle \mathcal{T}, \mathcal{S}, \mathcal{M}_1 \rangle$ and $\langle \mathcal{T}, \mathcal{S}, \mathcal{M}_2 \rangle$ are equivalent, i.e, $Mod(\langle \mathcal{T}, \mathcal{S}, \mathcal{M}_1 \rangle, D) = Mod(\langle \mathcal{T}, \mathcal{S}, \mathcal{M}_2 \rangle, D)$, for every legal instance D for $\mathcal{S}$. Hereinafter, for ease of exposition, we omit quantifiers in mapping assertions.

In this paper we use the following notion of inconsistency [Lembo *et al.*, 2015].

Definition 1 (Global mapping inconsistency)**.** *Let $\mathcal{T}$ be a TBox, $\mathcal{S}$ a source schema, and $\mathcal{M}$ a mapping. We say that $\mathcal{M}$ is globally inconsistent for $\langle \mathcal{T}, \mathcal{S} \rangle$ if no instance D legal for $\mathcal{S}$ exists such that $\mathcal{M}$ is active on D and $Mod(\mathcal{J}, D) \neq \emptyset$.*

Intuitively, it is impossible to activate simultaneously all the assertions of a globally inconsistent mapping without contradicting $\mathcal{T}$ and $\mathcal{S}$. Note that a classical notion of inconsistency considers, instead, also source instances that do not activate all assertions in $\mathcal{M}$, which means considering as consistent also specifications that admit models only if $D = \emptyset$, which is a trivial case for OBDA. Definition 1 recognizes the central role of the mapping, and is particularly suited in OBDA.

3 Mapping Repairs

In this section we define two operators for updating an OBDA specification when its TBox and/or source schema change. Each such operator is based on a specific notion of mapping repair. We start with *deletion-based mapping repairs*, where one looks for repairs among subsets of the original mapping, and then introduce *entailment-based mapping repairs*, aimed at preserving as much as possible of mapping assertions implied by the original specification.

From now on, for ease of exposition and without loss of generality, we assume that $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ is such that different mapping assertions in $\mathcal{M}$ use different variable symbols.

Below we provide our first definition of mapping repair.

Definition 2 (Deletion-based mapping repair). *Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ be an OBDA specification such that $\mathcal{M}$ is globally consistent for $\langle \mathcal{T}, \mathcal{S} \rangle$, $\mathcal{T}'$ a consistent TBox, $\mathcal{S}'$ a consistent source schema, and $\mathcal{M}'$ a mapping such that $\mathcal{M}' \subseteq \mathcal{M}$. We say that $\mathcal{M}'$ is a* deletion-based mapping repair (DMR) *for $\mathcal{J}$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$ if:*

1. *$\mathcal{M}'$ is globally consistent for $\langle \mathcal{T}', \mathcal{S}' \rangle$, and*

2. *there exists no mapping $\mathcal{M}'' \subseteq \mathcal{M}$ such that:*

    (a) *$\mathcal{M}''$ is globally consistent for $\langle \mathcal{T}', \mathcal{S}' \rangle$, and*
    (b) *$\mathcal{M}'' \supset \mathcal{M}'$.*

In other words, a DMR is a maximal subset of $\mathcal{M}$ that is globally consistent for the new TBox and source schema. It is easy to see that a DMR for $\mathcal{J}$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$ always exists, and that if $\mathcal{M}$ is finite, also $\mathcal{M}'$ is finite. Furthermore, if $\mathcal{M}$ is globally consistent for $\langle \mathcal{T}', \mathcal{S}' \rangle$, then $\mathcal{M}$ is the only DMR for $\mathcal{J}$. In general, however, several repairs exist.

We are now able to introduce our DMR-based update operator, denoted with $\bullet$, and define the OBDA specifications computed by such operator. Given $\mathcal{J}$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$, the set of such specifications, denoted by $\mathcal{J} \bullet \langle \mathcal{T}', \mathcal{S}' \rangle$, is

$$\{\langle \mathcal{T}', \mathcal{S}', \mathcal{M}' \rangle \mid \mathcal{M}' \text{ is a DMR for } \mathcal{J} \text{ under update } \langle \mathcal{T}', \mathcal{S}' \rangle\}.$$

Note that the above notions are actually independent of the initial TBox $\mathcal{T}$ and schema $\mathcal{S}$. Therefore, in the following we will simply call $\mathcal{M}'$ a DMR for $\mathcal{M}$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$, and will denote the set $\mathcal{J} \bullet \langle \mathcal{T}', \mathcal{S}' \rangle$ as $\mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S}' \rangle$.

Example 3. Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ be as follows:

$$\mathcal{T} = \{C \sqsubseteq F, \; C \sqsubseteq A, \; \exists P \sqsubseteq E, \; A \sqsubseteq E\} \quad \mathcal{S} = \{T\}$$
$$\mathcal{M} = \{T(x,y) \rightsquigarrow P(x,y), \; T(x',y') \rightsquigarrow C(x')\}.$$

where A, C, F, E are atomic concepts, P is an atomic role, and T is a binary relation. Moreover, let $\mathcal{T}' = \mathcal{T} \cup \{A \sqsubseteq \neg \exists P\}$. It is easy to see that $\mathcal{M}$ is not globally consistent for $\langle \mathcal{T}', \mathcal{S} \rangle$, since for every source schema that activates $\mathcal{M}$, the mapping produces two facts of the form $P(x,y)$ and $C(x)$ which violate the assertion $C \sqsubseteq \neg \exists P$, inferred by $\mathcal{T}'$. Then, $\mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S} \rangle = \{\langle \mathcal{T}', \mathcal{S}, \mathcal{M}_1 \rangle, \langle \mathcal{T}', \mathcal{S}, \mathcal{M}_2 \rangle\}$, where: $\mathcal{M}_1 = \{T(x,y) \rightsquigarrow P(x,y)\}$ and $\mathcal{M}_2 = \{T(x',y') \rightsquigarrow C(x')\}$. $\square$

The following definition formalizes the notion of query entailment in the DMR-based update framework.

Definition 4 (Query Entailment under DMR). *Let $\mathcal{M}$ be a mapping, $\mathcal{T}'$ a consistent TBox, $\mathcal{S}'$ a consistent source schema, D a legal instance for $\mathcal{S}'$, and q a BCQ. We say*

that q is entailed under DMR by $\mathcal{M}$, $\mathcal{T}'$, $\mathcal{S}'$, and D, denoted as $(\mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S}' \rangle, D) \models q$, if $(\mathcal{J}', D) \models q$ for every $\mathcal{J}' \in \mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S}' \rangle$.

Let $\mathcal{M}$, $\mathcal{S}$ and $\mathcal{T}'$ be as in Example 3, and let $D = \{T(a,b)\}$. It is easy to see that $(\mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S} \rangle, D) \models E(a)$.

We finally notice that DMRs depend on the syntax of the initial mapping $\mathcal{M}$. As the following example shows, two equivalent (but syntactically different) mappings may have non-equivalent DMRs under the same update.

Example 5. Let $\mathcal{T}$, $\mathcal{S}$, $\mathcal{M}$ and $\mathcal{T}'$ be as in Example 3, and let $\mathcal{M}'$ be the mapping $\{T(x,y) \rightsquigarrow P(x,y), T(x',y') \rightsquigarrow C(x'), T(x'',y'') \rightsquigarrow F(x'')\}$. $\mathcal{M}$ and $\mathcal{M}'$ are equivalent with respect to $\mathcal{T}$, since $\mathcal{T} \models C \sqsubseteq F$. However, while $\mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S} \rangle$ is as in Example 3, $\mathcal{M}' \bullet \langle \mathcal{T}', \mathcal{S} \rangle = \{\langle \mathcal{T}', \mathcal{S}, \mathcal{M}'_1 \rangle, \langle \mathcal{T}', \mathcal{S}, \mathcal{M}'_2 \rangle\}$, where: $\mathcal{M}'_1 = \{T(x,y) \rightsquigarrow P(x,y), T(x'',y'') \rightsquigarrow F(x'')\}$ and $\mathcal{M}'_2 = \{T(x',y') \rightsquigarrow C(x'), T(x'',y'') \rightsquigarrow F(x'')\}$. This implies that query entailment is different in the two cases. E.g., given the source instance $D = \{T(a,b)\}$, we have that $(\mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S} \rangle, D) \not\models F(a)$, while $(\mathcal{M}' \bullet \langle \mathcal{T}', \mathcal{S} \rangle, D) \models F(a)$. $\square$

We also notice that DMRs do not preserve mapping assertions not included in the original specification but entailed by it, and that would not directly lead to a contradiction with the update. For instance, in Example 3 the mapping $T(x'',y'') \rightsquigarrow F(x'')$ is implied by $\mathcal{J}$ and is not in contradiction with other mapping assertions under $\mathcal{T}'$, but it is not entailed in $\langle \mathcal{T}', \mathcal{S}, \mathcal{M}_1 \rangle$.

In the following, we provide a different definition of mapping repair, which tries to preserve entailed mappings, and guarantees a reasonable form of syntax independence.

We first introduce some preliminary definitions.

Definition 6 (Mapping entailment set). *Given an OBDA specification $\mathcal{J}$ and a mapping language $\mathcal{L}$, the* mapping entailment set (MES) *of $\mathcal{J}$ in $\mathcal{L}$, denoted by $MES_{\mathcal{L}}(\mathcal{J})$, is the set of mapping assertions $\{m \in \mathcal{L} \mid \mathcal{J} \models m$ and there exists $m_0 \in \mathcal{M}$ such that $body(m_0) = body(m)\}$.*

Among mapping assertions implied by $\mathcal{J}$, in the above definition we excluded those using queries in their bodies that are different than those already present in $\mathcal{M}$. Indeed, we cannot assume that any possible query over $\mathcal{S}$ can be really asked to the data sources, since usually they allow only for a controlled access to their content. We thus prefer to consider only queries already in the specification, which we can assume to be of the form accepted at the sources.

Note that, if the mapping language is infinite, the mapping entailment set $MES_{\mathcal{L}}(\mathcal{J})$ may be an infinite set of assertions even if the mapping $\mathcal{M}$ is a finite set.

We now need to give a characterization of the notion of "closeness" between sets. To this aim, we refer to the *fewer changes* relation given in [Fagin *et al.*, 1983].

Let $\mathcal{M}$, $\mathcal{M}_1$, and $\mathcal{M}_2$ be three sets. We say that $\mathcal{M}_1$ has *fewer deletions* than (resp. *same deletions of*) $\mathcal{M}_2$ with respect to $\mathcal{M}$ if $\mathcal{M} \setminus \mathcal{M}_1 \subset \mathcal{M} \setminus \mathcal{M}_2$ (resp. $\mathcal{M} \setminus \mathcal{M}_1 = \mathcal{M} \setminus \mathcal{M}_2$). We also say that $\mathcal{M}_1$ has *fewer insertions* than $\mathcal{M}_2$ with respect to $\mathcal{M}$ if $\mathcal{M}_1 \setminus \mathcal{M} \subset \mathcal{M}_2 \setminus \mathcal{M}$.

Definition 7 (Fewer changes). *Let $\mathcal{M}$, $\mathcal{M}_1$, and $\mathcal{M}_2$ be three sets. We say that $\mathcal{M}_1$ has* fewer changes *than $\mathcal{M}_2$ with respect to $\mathcal{M}$ if:*

1. $\mathcal{M}_1$ *has fewer deletions than* $\mathcal{M}_2$ *with respect to* $\mathcal{M}$*; or*

2. $\mathcal{M}_1$ *and* $\mathcal{M}_2$ *have the same deletions with respect to* $\mathcal{M}$*, and* $\mathcal{M}_1$ *has fewer insertions than* $\mathcal{M}_2$ *with respect to* $\mathcal{M}$*.*

Notice that the notion of *fewer changes* gives preference to the sets having fewer deletions over those having fewer insertions. Such behavior is justified by the idea of retaining as much information as possible from the original set $\mathcal{M}$.

We can now provide a new definition of mapping repair.

Definition 8 (Entailment-based mapping repair)**.** *Let* $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ *be an OBDA specification such that* $\mathcal{M}$ *is globally consistent for* $\langle \mathcal{T}, \mathcal{S} \rangle$*,* $\mathcal{T}'$ *a consistent TBox,* $\mathcal{S}'$ *a consistent source schema,* $\mathcal{L}$ *a mapping language, and* $\mathcal{M}'$ *an* $\mathcal{L}$*-mapping. We say that* $\mathcal{M}'$ *is an* entailment-based $\mathcal{L}$*-mapping repair (* $\mathcal{L}$*-EMR) for* $\mathcal{J}$ *under update* $\langle \mathcal{T}', \mathcal{S}' \rangle$ *if:*

1. $\mathcal{M}'$ *is globally consistent for* $\langle \mathcal{T}', \mathcal{S}' \rangle$*; and*

2. *there exists no* $\mathcal{L}$*-mapping* $\mathcal{M}''$ *such that:*

    (a) $\mathcal{M}''$ *is globally consistent for* $\langle \mathcal{T}', \mathcal{S}' \rangle$*; and*
    (b) $MES_\mathcal{L}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}'' \rangle)$ *has fewer changes than* $MES_\mathcal{L}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}' \rangle)$ *with respect to* $MES_\mathcal{L}(\mathcal{J})$*.*

Intuitively, an entailment-based mapping repair (or more precisely its MES with respect to $\mathcal{T}'$ and $\mathcal{S}'$) preserves as much as possible of the mappings that were entailed in the initial OBDA specification and that do not contradict the update.

The following proposition shows the relationship between $\mathcal{L}$-EMRs and DMRs of $MES_\mathcal{L}(\mathcal{J})$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$.

Proposition 9. *For every* $\mathcal{L}$*-EMR* $\mathcal{M}'$ *for* $\mathcal{J}$ *under update* $\langle \mathcal{T}', \mathcal{S}' \rangle$ *there exists a DMR* $\mathcal{M}''$ *for* $MES_\mathcal{L}(\mathcal{J})$ *under update* $\langle \mathcal{T}', \mathcal{S}' \rangle$ *such that* $MES_\mathcal{L}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}' \rangle) = MES_\mathcal{L}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}'' \rangle)$*, and vice-versa, i.e., for every DMR* $\mathcal{M}''$ *for* $MES_\mathcal{L}(\mathcal{J})$ *under update* $\langle \mathcal{T}', \mathcal{S}' \rangle$ *there exists a* $\mathcal{L}$*-EMR* $\mathcal{M}'$ *for* $\mathcal{J}$ *under update* $\langle \mathcal{T}', \mathcal{S}' \rangle$ *such that* $MES_\mathcal{L}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}'' \rangle) = MES_\mathcal{L}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}' \rangle)$*.*

We notice that, differently from DMRs, EMRs depend on the language $\mathcal{L}$ in which the repair is specified. This is due to the fact that EMRs may also contain mapping assertions that are not contained in $\mathcal{M}$, and we thus need to specify the language in which such assertions are expressed.

Example 10. Let $\mathcal{J}$ and $\mathcal{T}'$ be as in Example 3. One can easily verify that the mappings $\mathcal{M}'_1 = \{T(x,y) \leadsto P(x,y),\ T(x',y') \leadsto F(x')\}$ and $\mathcal{M}'_2 = \{T(x,y) \leadsto C(x)\}$ are GAV-EMRs. Moreover, for any other GAV-EMR $\mathcal{M}'_3$, we have that $MES_{GAV}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}'_3 \rangle) = MES_{GAV}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}'_1 \rangle)$ or $MES_{GAV}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}'_3 \rangle) = MES_{GAV}(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}'_2 \rangle)$. $\square$

Similarly to DMR, we now introduce an operator, denoted $\circ_\mathcal{L}$, to update an OBDA specification through EMR. Then, the set of OBDA specifications computed by $\circ_\mathcal{L}$ for $\mathcal{J}$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$, denoted $\mathcal{J} \circ_\mathcal{L} \langle \mathcal{T}', \mathcal{S}' \rangle$, is

$$\{\langle \mathcal{T}', \mathcal{S}', \mathcal{M}' \rangle \mid \mathcal{M}' \text{ is a } \mathcal{L}\text{-EMR for } \mathcal{J} \text{ under update} \langle \mathcal{T}', \mathcal{S}' \rangle\}.$$

Consider now again Example 5, and the equivalent specifications $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ and $\mathcal{J}' = \langle \mathcal{T}, \mathcal{S}, \mathcal{M}' \rangle$. It is easy to see that $\mathcal{J} \circ_{GAV} \langle \mathcal{T}', \mathcal{S} \rangle = \mathcal{J}' \circ_{GAV} \langle \mathcal{T}', \mathcal{S} \rangle$, i.e., despite the syntactic difference between $\mathcal{M}$ and $\mathcal{M}'$, the set

of GAV-EMRs is the same in the two cases (and so are the sets of OBDA specifications in the two cases). We point out that syntax independence for EMRs does not hold in general. However, it can be easily shown that two equivalent OBDA specifications $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ and $\mathcal{J}' = \langle \mathcal{T}, \mathcal{S}, \mathcal{M}' \rangle$ that are not *mapping redundant* (i.e., deleting any mapping assertion from $\mathcal{J}$ or $\mathcal{J}'$ produces a non-equivalent specification) have always equivalent EMRs with respect to any update. This does not hold for DMRs.

Below we give the notion of query entailment for $\mathcal{L}$-EMRs.

Definition 11 (Query Entailment under $\mathcal{L}$-EMR)**.** *Let* $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ *be an OBDA specification such that* $\mathcal{M}$ *is globally consistent for* $\langle \mathcal{T}, \mathcal{S} \rangle$*,* $\mathcal{T}'$ *a consistent TBox,* $\mathcal{S}'$ *a consistent source schema,* $\mathcal{L}$ *a mapping language,* D *a legal instance for* $\mathcal{S}'$*, and* q *a BCQ. We say that* q *is entailed under* $\mathcal{L}$*-EMR by* $\mathcal{J}, \mathcal{T}', \mathcal{S}'$*, and* D*, denoted as* $(\mathcal{J} \circ_\mathcal{L} \langle \mathcal{T}', \mathcal{S}' \rangle, D) \models q$*, if* $(\mathcal{J}', D) \models q$ *for every* $\mathcal{J}' \in \mathcal{J} \circ_\mathcal{L} \langle \mathcal{T}', \mathcal{S}' \rangle$*.*

Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ and $\mathcal{T}'$ as in Example 3, and let $D = \{T(a,b)\}$. It is easy to see that $(\mathcal{J} \circ_{GAV} \langle \mathcal{T}', \mathcal{S} \rangle, D) \models F(a)$, while $(\mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S} \rangle, D) \not\models F(a)$.

We finally note that, differently from DMRs, a *finite* $\mathcal{L}$-EMR does not always exist, even if $\mathcal{M}$ is finite, thus the study of this case is particularly challenging.

4 Query Entailment under DMRs

In this section we study the query entailment problem under deletion-based repairs. First, we define the following general algorithm for checking whether $(\mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S}' \rangle, D) \models q$, whose correctness follows immediately from Definition 2.

Algorithm 1: CheckDMR-QE $(\mathcal{M}, \mathcal{T}', \mathcal{S}', D, q)$

Input: finite GLAV mapping $\mathcal{M}$, TBox $\mathcal{T}'$, source schema $\mathcal{S}'$ legal instance D for $\mathcal{S}'$, BCQ q over $\mathcal{T}'$
if there exists a subset $\mathcal{M}'$ of $\mathcal{M}$ such that

  (i) $\mathcal{M}'$ is globally consistent for $\langle \mathcal{T}', \mathcal{S}' \rangle$ **and**
  (ii)   for every $m \in \mathcal{M} \setminus \mathcal{M}'$, $\mathcal{M}' \cup \{m\}$ is globally inconsistent for $\langle \mathcal{T}', \mathcal{S}' \rangle$ **and**
  (iii) $(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}' \rangle, D) \not\models q$

then return *false*
else return *true*

We now focus on the setting with *DL-Lite$_R$* TBoxes and simple source schemas[1], and provide the following result.

Theorem 12. *Let* $\mathcal{T}'$ *be a DL-Lite$_R$ TBox,* $\mathcal{S}'$ *a simple source schema,* $\mathcal{M}$ *a finite GLAV mapping,* q *a BCQ over* $\mathcal{T}'$*, and* D *a legal instance for* $\mathcal{S}'$*. Deciding* $(\mathcal{M} \bullet \langle \mathcal{T}', \mathcal{S}' \rangle, D) \models q$ *is* Π_2^p*-complete in combined complexity and in* AC^0 *in data complexity.*

Proof (sketch). The complexity of Algorithm CheckDMR-QE is coNP using an oracle that is able to decide global mapping inconsistency and CQ entailment. Since in this case both such problems are NP-complete in combined complexity when the TBox is *DL-Lite$_R$* [Lembo *et al.*, 2015], we obtain an upper bound of Π_2^p. Through a reduction from 2-QBF validity we can in fact show that the above bound is exact.

---

[1]Notice that *DL-Lite$_R$* TBoxes and simple source schemas are always consistent, as required by Definition 2.

As for data complexity, all DMRs for a mapping $\mathcal{M}$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$ can be computed in constant time (through steps (i) and (ii) of CheckDMR-QE), since this task does not depend on the data. Then, the data complexity coincides with that of checking $(\langle \mathcal{T}', \mathcal{S}', \mathcal{M}' \rangle, D) \models q$ for each DMR $\mathcal{M}'$, which is AC^0 in data complexity [Calvanese *et al.*, 2009] □

5 Query Entailment under EMRs in GAV

We focus on the case when TBoxes are expressed in *DL-Lite$_R$*, mappings are GAV, and the source schemas are simple.

First, we need to introduce the notion of *mapping saturation with respect to a TBox*. In the definition, C and F denote atomic concepts, and P and S denote atomic roles.

Definition 13. *Let $\mathcal{T}$ be a DL-Lite$_R$ TBox and $\mathcal{M}$ a GAV mapping. We define the* saturation *of $\mathcal{M}$ with respect to $\mathcal{T}$, denoted by $sat_{GAV}^{\mathcal{T}}(\mathcal{M})$, as follows:*

- *if $\phi(x) \rightarrow C(x)$ is in $\mathcal{M}$, then $\phi(x) \rightarrow F(x)$ is in $sat_{GAV}^{\mathcal{T}}(\mathcal{M})$, for every F such that $\mathcal{T} \models C \sqsubseteq F$;*
- *if $\phi(x, y) \rightarrow P(x, y)$ is in $\mathcal{M}$, then:*
  - *$\phi(x, y) \rightarrow C(x)$ (resp., $\phi(x, y) \rightarrow C(y)$) is in $sat_{GAV}^{\mathcal{T}}(\mathcal{M})$, for every C such that $\mathcal{T} \models \exists P \sqsubseteq C$ (resp., $\mathcal{T} \models \exists P^- \sqsubseteq C$);*
  - *$\phi(x, y) \rightarrow S(x, y)$ (resp., $\phi(x, y) \rightarrow S(y, x)$) is in $sat_{GAV}^{\mathcal{T}}(\mathcal{M})$, for every S such that $\mathcal{T} \models P \sqsubseteq S$ (resp., $\mathcal{T} \models P \sqsubseteq S^-$);*

We note that our notion of mapping saturation is equivalent to that of $\mathcal{T}$-mapping given in [Rodríguez-Muro and Calvanese, 2011]. Roughly, the set $sat_{GAV}^{\mathcal{T}}(\mathcal{M})$ represents a set of mapping assertions (which is finite if $\mathcal{M}$ is finite) that are relevant for computing GAV-EMRs, as stated below.

Theorem 14. *Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$, with $\mathcal{T}$ a DL-Lite$_R$ TBox, $\mathcal{S}$ a simple source schema, and $\mathcal{M}$ a GAV mapping globally consistent for $\langle \mathcal{T}, \mathcal{S} \rangle$, and let $\mathcal{T}'$ be a DL-Lite$_R$ TBox and $\mathcal{S}'$ a simple source schema. For every GAV-EMR $\mathcal{M}'$ for $\mathcal{J}$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$ there exists a DMR $\mathcal{M}''$ for $sat_{GAV}^{\mathcal{T}}(\mathcal{M})$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$ such that $\mathcal{M}'$ and $\mathcal{M}''$ are equivalent with respect to $\mathcal{T}'$, and vice-versa, i.e., for every DMR $\mathcal{M}''$ for $sat_{GAV}^{\mathcal{T}}(\mathcal{M})$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$ there exists a GAV-EMR $\mathcal{M}'$ for $\mathcal{J}$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$ such that $\mathcal{M}''$ and $\mathcal{M}'$ are equivalent with respect to $\mathcal{T}'$.*

Proof (sketch). The thesis follows from Proposition 9 and the fact that for each mapping $m' \in MES_{GAV}(\mathcal{J})$ there exists a mapping $m'' \in sat_{GAV}^{\mathcal{T}}(\mathcal{M})$ such that $head(m') = head(m'')$ and $body(m') \subseteq body(m'')$. □

A direct consequence of the above theorem is that we can reduce query entailment under GAV-EMR to query entailment under DMR, i.e., to decide entailment of a BCQ q under GAV-EMR we can execute CheckDMR-QE $(sat_{GAV}^{\mathcal{T}}(\mathcal{M}), \mathcal{T}', \mathcal{S}', D, q)$, where $\mathcal{M}$, $\mathcal{T}'$, and $\mathcal{S}'$ are as in Theorem 14, and D is a legal instance for $\mathcal{S}$. The above result immediately implies the following theorem.

Theorem 15. *Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ be an OBDA specification, where $\mathcal{T}$ is a DL-Lite$_R$ TBox, $\mathcal{S}$ is a simple source schema, and $\mathcal{M}$ is a finite GAV mapping that is globally consistent for $\langle \mathcal{T}, \mathcal{S} \rangle$. Let $\mathcal{T}'$ be a DL-Lite$_R$ TBox, $\mathcal{S}'$ a simple source*

schema, D a legal instance for $\mathcal{S}'$, and q a BCQ over $\mathcal{T}'$. Deciding $(\mathcal{J} \circ_{GAV} \langle \mathcal{T}', \mathcal{S}' \rangle, D) \models q$ is Π_2^p-complete in combined complexity and in AC^0 in data complexity.

Proof (sketch). To prove the theses we exploit Theorem 12 and the fact that $sat_{GAV}^{\mathcal{T}}(\mathcal{M})$ can be computed in polynomial time w.r.t. the size of $\mathcal{T}$ and $\mathcal{M}$ and is independent of D. □

6 Query Entailment under EMRs in GLAV

As stated by Proposition 9, we can decide CQ entailment under EMRs for GLAV mappings by computing the DMRs of $MES_{GLAV}(\mathcal{J})$. In order to be effective, such a reduction of EMR query entailment to DMR query entailment requires to overcome the problem of dealing with the infinite mapping $MES_{GLAV}(\mathcal{J})$. In the following, our main goal is to identify a finite subset of $MES_{GLAV}(\mathcal{J})$ that is correct for the purpose of CQ entailment under EMRs for GLAV mappings and *DL-Lite$_R$* TBoxes. We reach this goal through the notions of *mapping chase* (*MapChase*) and *mapping expansion* (*MapExp*). This allows us to prove the decidability of CQ entailment under EMRs and establish its data complexity under a condition on the negative role inclusions of the updated TBox.

From now on, we assume that $\mathcal{T}$ and $\mathcal{T}'$ are *DL-Lite$_R$* TBoxes, and we also assume without loss of generality that $\mathcal{T}$ and $\mathcal{T}'$ have the same signature. Moreover, we denote by $\mathcal{T}^*$ the positive deductive closure of $\mathcal{T}$, i.e., the set of positive concept and role inclusions entailed by $\mathcal{T}$. Finally, we denote by $NR(\mathcal{T})$ the negative role inclusions entailed by $\mathcal{T}$.

We now adapt the well-known *chase* of an ABox with respect to a set of *DL-Lite$_R$* positive inclusions [Calvanese *et al.*, 2007] to the heads of GLAV mapping assertions. We call *atom* a GLAV mapping head atom, i.e., an expression of the form $C(x)$ or $P(x, x')$, where C is an atomic concept, P is an atomic role, and x, x' are variable symbols. Given a *DL-Lite$_R$* positive inclusion I, an atom α, and a set of atoms Φ such that $\alpha \in \Phi$, we denote by $Res(\alpha, I, \Phi)$ the atom computed by the application of I to α, similar to chase computation for *DL-Lite* ontologies. For example, if $\alpha = C(x)$ and $I = C \sqsubseteq \exists P$, $Res(\alpha, I, \Phi) = P(x, z)$, where z is a fresh variable in Φ.

Given a set of atoms Φ and a *DL-Lite$_R$* TBox $\mathcal{T}$, we define:

- $chase^0(\Phi, \mathcal{T}) = \Phi$;
- for every integer k such that $k \geq 1$, $chase^k(\Phi, \mathcal{T}) = chase^{k-1}(\Phi, \mathcal{T}) \cup \{\alpha \mid \beta \in chase^{k-1}(\Phi, \mathcal{T}) \text{ and } I \in \mathcal{T} \text{ and } \alpha = Res(\beta, I, chase^{k-1}(\Phi, \mathcal{T}))\}$.

We now define a notion of chase for GLAV mapping assertions in *DL-Lite$_R$*. Given an OBDA specification $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ and an integer $k \geq 0$, we define *MapChase^k*$(\mathcal{J})$ as the set of mapping assertions of the form $body(m) \rightsquigarrow H$, such that $m \in \mathcal{M}$ and $\emptyset \subset H \subseteq chase^k(head(m), \mathcal{T}^*)$. Moreover, we define *MapChase*$(\mathcal{J}) = \bigcup_{k=0}^{\infty}$ *MapChase^k*$(\mathcal{J})$.

Informally, *MapChase*$(\mathcal{J})$ augments the initial mapping $\mathcal{M}$ by adding mapping assertions that are obtained expanding (through the chase w.r.t. $\mathcal{T}$) the heads of the assertions in $\mathcal{M}$.

We say that a GLAV mapping $\mathcal{M}$ is *body-saturated* if, for every pair of mapping assertions $m_1, m_2 \in \mathcal{M}$, if there exists a homomorphism $h : Vars(body(m_1)) \rightarrow Vars(body(m_2))$ (where $Vars(\Phi)$ denotes the set of variables occurring in Φ)

such that $h(body(m_1)) \subseteq body(m_2)$, then the mapping assertion $body(m_2) \rightsquigarrow head(m_2), h(head(m_1))$ belongs to $\mathcal{M}$.

For body-saturated mappings, the following property holds.

Theorem 16. *Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ where $\mathcal{M}$ is a body-saturated mapping. For every $m \in MES_{GLAV}(\mathcal{J})$ there exists $m' \in MapChase(\mathcal{J})$ such that $body(m') = body(m)$ and there exists a homomorphism $h : NFVars(head(m)) \rightarrow Vars(head(m'))$ such that $h(head(m)) \subseteq head(m')$, where $NFVars(head(m))$ denotes the set of non-frontier variables occurring in $head(m)$.*

From now on, we assume without loss of generality that $\mathcal{M}$ is body-saturated. Indeed, if $\mathcal{M}$ violates this assumption, it is possible to construct in finite time an equivalent body-saturated GLAV mapping (by adding a finite number of mapping assertions to $\mathcal{M}$). Note that every mapping assertion in $MapChase(\mathcal{J})$ belongs to $MES_{GLAV}(\mathcal{J})$, while the opposite does not hold. Furthermore, it can easily be verified that it is not correct to use $MapChase(\mathcal{J})$ instead of $MES_{GLAV}(\mathcal{J})$ to compute the EMRs of $\mathcal{J}$ under some update $\langle \mathcal{T}', \mathcal{S}' \rangle$. To reach such a property, we have to expand the set $MapChase(\mathcal{J})$.

Let Φ be a set of atoms. For every positive integer k, we define $AtomExp^k(\Phi)$ as the set of atoms

$$\{C(z) \mid C(x) \in \Phi \text{ and } z \in \{x, x^1, \dots, x^k\}\} \cup$$
$$\{R(z, w) \mid R(x, y) \in \Phi \text{ and } z \in \{x, x^1, \dots, x^k\}$$
$$\text{and } w \in \{y, y^1, \dots, y^k\}\}$$

Moreover, we define $AtomExp(\Phi) = \bigcup_{k=1}^{\infty} AtomExp^k(\Phi)$.

Intuitively, $AtomExp(\Phi)$ expands Φ with "copies" of the atoms of Φ, using "copies" x^i of each variable x occurring in Φ. It is immediate to see that such an expansion of the set of atoms Φ does not change the meaning of Φ (interpreted as an existentially quantified conjunction of atoms).

We now define a notion of expansion of a mapping $\mathcal{M}$ that applies the function $AtomExp$ to the heads of the assertions in $\mathcal{M}$. Let m be a GLAV mapping assertion. We define $MapExp^k(m)$ as the set $\{m' \mid body(m') = body(m) \text{ and } \emptyset \subset head(m') \subseteq AtomExp^k(head(m))\}$. Given a GLAV mapping $\mathcal{M}$, we define $MapExp^k(\mathcal{M}) = \bigcup_{m \in \mathcal{M}} MapExp^k(m)$. Finally, we define $MapExp(\mathcal{M}) = \bigcup_{k=1}^{\infty} MapExp^k(\mathcal{M})$.

Theorem 17. *Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$. Then: (i) $MapExp(MapChase(\mathcal{J})) \subseteq MES_{GLAV}(\mathcal{J})$; (ii) for every $m \in MES_{GLAV}(\mathcal{J})$ there exists $m' \in MapExp(MapChase(\mathcal{J}))$ such that m' is equal to m up to renaming of the non-frontier variables of $head(m')$.*

Proof (sketch). Thesis (i) follows trivially from the definitions of $MapChase$ and $MapExp$. Thesis (ii) follows from Theorem 16 and from the key property that, for every mapping $m \in \mathcal{M}$ and for every $m' \in MES_{GLAV}(\mathcal{J})$ such that $body(m') = body(m)$ and there exists a homomorphism $h : NFVars(head(m')) \rightarrow Vars(head(m))$ such that $h(head(m)) \subseteq head(m')$, the set of atoms $head(m')$ is a subset of $MapExp(chase(head(m), \mathcal{T}))$ (up to renaming of the non-frontier variables of $head(m')$). □

Theorem 17 implies that $MapExp(MapChase(\mathcal{J}))$ is a subset of $MES_{GLAV}(\mathcal{J})$ that produces equivalent DMRs under any update. However, such a subset is still infinite, due to: (i) the infiniteness introduced by $MapChase$ (when $\mathcal{T}$ contains cycles of existentially quantified concepts); (ii) the inifiniteness

introduced by $MapExp$. We first deal with the latter problem, showing that we can safely restrict $MapExp$ to a finite number of expansion steps without losing the above property. More precisely, it is possible to prove that there exists k such that $MapExp^k(MapChase(\mathcal{J}))$ and $MapExp(MapChase(\mathcal{J}))$ have equivalent DMRs under arbitrary updates.

Theorem 18. *Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ be an OBDA specification, where $\mathcal{T}$ is a DL-Lite$_R$ TBox, $\mathcal{S}$ is a simple source schema, and $\mathcal{M}$ is a GLAV mapping, let $\mathcal{T}'$ be a DL-Lite$_R$ TBox such that $\mathcal{M}$ is globally consistent for $\langle \mathcal{T} \cup NR(\mathcal{T}'), \mathcal{S} \rangle$ and let $\mathcal{S}'$ be a simple source schema. There exists an integer k such that, for every DMR $\mathcal{M}'$ of $MES_{GLAV}(\mathcal{J})$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$ there exists a DMR $\mathcal{M}''$ of $MapExp^k(MapChase(\mathcal{J}))$ under update $\langle \mathcal{T}', \mathcal{S}' \rangle$ such that $\mathcal{M}'$ and $\mathcal{M}''$ are equivalent with respect to $\mathcal{T}'$.*

Based on the previous theorem, we are able to establish the data complexity of query entailment under EMR in GLAV for DL-Lite$_R$ TBoxes under the condition that the mapping $\mathcal{M}$ is consistent with the negative role inclusions of the updated TBox $\mathcal{T}'$, i.e., $\mathcal{M}$ is globally consistent for $\langle \mathcal{T} \cup NR(\mathcal{T}'), \mathcal{S} \rangle$.

Theorem 19. *Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{S}, \mathcal{M} \rangle$ with $\mathcal{T}$ DL-Lite$_R$ TBox, $\mathcal{S}$ simple source schema, and $\mathcal{M}$ finite GLAV mapping. Let $\mathcal{T}'$ be a DL-Lite$_R$ TBox such that $\mathcal{M}$ is globally consistent for $\langle \mathcal{T} \cup NR(\mathcal{T}'), \mathcal{S} \rangle$, $\mathcal{S}'$ a simple source schema, D a legal instance for $\mathcal{S}'$, and q a BCQ over $\mathcal{T}'$. Deciding $(\mathcal{J} \circ_{GLAV} \langle \mathcal{T}', \mathcal{S}' \rangle, D) \models q$ is in AC^0 in data complexity.*

It can be shown that, for $\mathcal{T}'$ such that $\mathcal{M}$ is not globally consistent for $\langle \mathcal{T} \cup NR(\mathcal{T}'), \mathcal{S} \rangle$, Theorem 18 no longer holds, and as a consequence Theorem 19 does not apply as well.

7 Conclusions

Besides their theoretical relevance, the results presented in this paper have a practical fallout. Typically, in ontology (and thus in OBDA) design a set of so-called competency questions is used, which are representative of the intended usages of the system. The ability of successfully answering such questions is a measure of the quality of the design, and in particular of the fitness of the system with respect to its functional requirements [Neuhaus and Vizedom, 2013]. Therefore, query answering under our repair semantics can be of help for the maintenance of an evolving OBDA system, since it can allow the designer to measure the impact of repairing with respect to the ability of the system to answer competency questions. We also note that variants of this service might be considered, for instance, to identify repairs under which the system has a desired behaviour with respect to query answering. We believe this is a promising direction towards semi-automatic mapping evolution management, even though several aspects still need to be investigated. In particular, we want to complete the study of CQ entailment under EMRs for GLAV mappings for DL-Lite$_R$ ontologies and analyze query entailment under our repair semantics for different DLs. Furthermore, we aim at defining approximations of the notions of DMR and EMR that guarantee the existence of a single repair, which might have important practical implications.

References

[Abiteboul *et al.*, 1995] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison Wesley Publ. Co., 1995.

[Antonioli *et al.*, 2014] Natalia Antonioli, Francesco Castanò, Spartaco Coletta, Stefano Grossi, Domenico Lembo, Maurizio Lenzerini, Antonella Poggi, Emanuela Virardi, and Patrizia Castracane. Ontology-based data management for the italian public debt. In *Proc. of the 8th Int. Conf. on Formal Ontology in Information Systems (FOIS 2014)*, pages 372–385, 2014.

[Baader *et al.*, 2007] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2nd edition, 2007.

[Calvanese *et al.*, 2007] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. of Automated Reasoning*, 39(3):385–429, 2007.

[Calvanese *et al.*, 2009] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Riccardo Rosati, and Marco Ruzzi. Using OWL in data integration. In *Semantic Web Information Management - A Model-Based Perspective*, pages 397–424. Springer, 2009.

[Das *et al.*, 2012] Souripriya Das, Seema Sundara, and Richard Cyganiak. R2RML: RDB to RDF Mapping Language. *W3C RDB2RDF Working Group*, W3C recommendation, September 2012.

[Doan *et al.*, 2012] AnHai Doan, Alon Y. Halevy, and Zachary G. Ives. *Principles of Data Integration*. Morgan Kaufmann, 2012.

[Fagin *et al.*, 1983] Ronald Fagin, Jeffrey D. Ullman, and Moshe Y. Vardi. On the semantics of updates in databases. In *Proc. of the 2nd ACM SIGACT SIGMOD Symp. on Principles of Database Systems (PODS'83)*, pages 352–365, 1983.

[Giese *et al.*, 2015] Martin Giese, Ahmet Soylu, Guillermo Vega-Gorgojo, Arild Waaler, Peter Haase, Ernesto Jiménez-Ruiz, Davide Lanti, Martín Rezk, Guohui Xiao, Özgür L. Özçep, and Riccardo Rosati. Optique: Zooming in on big data. *IEEE Computer*, 48(3):60–67, 2015.

[Hitzler *et al.*, 2012] Pascal Hitzler, Markus Krötzsch, Bijan Parsia, Peter F. Patel-Schneider, and Sebastian Rudolph. OWL 2 Web Ontology Language: Primer (second edition). W3C Recommendation, World Wide Web Consortium, December 2012. Available at http://www.w3.org/TR/owl2-primer/.

[Jiménez-Ruiz *et al.*, 2013] Ernesto Jiménez-Ruiz, Christian Meilicke, Bernardo Cuenca Grau, and Ian Horrocks. Evaluating mapping repair systems with large biomedical ontologies. In *Proc. of the 26th Int. Workshop on Description Logic (DL 2013)*, volume 1014 of *CEUR Electronic Workshop Proceedings,* http://ceur-ws.org/, pages 246–257, 2013.

[Lembo *et al.*, 2014] Domenico Lembo, José Mora, Riccardo Rosati, Domenico Fabio Savo, and Evgenij Thorstensen. Towards mapping analysis in ontology-based data access. In *Proc. of the 8th Int. Conf. on Web Reasoning and Rule Systems (RR 2014)*, pages 108–123, 2014.

[Lembo *et al.*, 2015] Domenico Lembo, José Mora, Riccardo Rosati, Domenico Fabio Savo, and Evgenij Thorstensen. Mapping analysis in ontology-based data access: Algorithms and complexity. In *Proc. of the 14th Int. Semantic Web Conf. (ISWC 2015)*, pages 217–234, 2015.

[Meilicke *et al.*, 2009] Christian Meilicke, Heiner Stuckenschmidt, and Andrei Tamilin. Reasoning support for mapping revision. *J. of Logic and Computation*, 19(5):807–829, 2009.

[Motik *et al.*, 2009] Boris Motik, Achille Fokoue, Ian Horrocks, Zhe Wu, Carsten Lutz, and Bernardo Cuenca Grau. OWL Web Ontology Language profiles. W3C Recommendation, World Wide Web Consortium, October 2009. Available at http://www.w3.org/TR/owl-profiles/.

[Neuhaus and Vizedom, 2013] Fabian Neuhaus and Amanda Vizedom. Ontology Summit 2013 Communiqué: Towards Ontology Evaluation across the Life Cycle, 2013. Available at http://ontolog.cim3.net/OntologySummit/2013/communique.html.

[Poggi *et al.*, 2008] Antonella Poggi, Domenico Lembo, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. Linking data to ontologies. *J. on Data Semantics*, X:133–173, 2008.

[Qi *et al.*, 2009] Guilin Qi, Qiu Ji, and Peter Haase. A conflict-based operator for mapping revision. In *Proc. of the 8th Int. Semantic Web Conf. (ISWC 2009)*, pages 521–536, 2009.

[Rahm and Bernstein, 2006] Erhard Rahm and Philip A. Bernstein. An online bibliography on schema evolution. *SIGMOD Record*, 35(4):30–31, 2006.

[Rodríguez-Muro and Calvanese, 2011] Mariano Rodríguez-Muro and Diego Calvanese. Dependencies: Making ontology based data access work in practice. In *Proc. of the 5th Alberto Mendelzon Int. Workshop on Foundations of Data Management (AMW 2011)*, volume 749 of *CEUR Electronic Workshop Proceedings,* http://ceur-ws.org/, 2011.

[Zablith *et al.*, 2015] Fouad Zablith, Grigoris Antoniou, Mathieu d'Aquin, Giorgos Flouris, Haridimos Kondylakis, Enrico Motta, Dimitris Plexousakis, and Marta Sabou. Ontology evolution: a process-centric survey. *Knowledge Eng. Review*, 30(1):45–75, 2015.